

20 May 1982 Vol 1 No 5

30p

Play Labyrinth on ZX81

Reviews: Pinball

Galaxy Invaders

Vic-20 disc drive

More on ZX Spectrum

Functions on BBC


BYTESHOP COMPUTERLAND
VIC
VIC 20
See it here, buy it now

POPULAR Computing WEEKLY

The Team

Editor
Duncan Scot

Reporter
David Kelly

Sub-editor
Peter Harvey

Designer
Eric Robbie

Editorial Secretary
Fiona McCormick

Advertisement Manager
David Lake [01-839 2846]

Advertisement Executive
Peter Chandler [01-839 1881]

Software Editor
Peter Gerrard [01-839 1855]

Publishing Director
Nick Hampshire

Popular Computing Weekly Hobhouse Court, 19 Whitcomb Street, London WC2
Telephone 01-839 6835

Published by Sunshine Publications Ltd

Typesetting origination and printing by Chesham Press Chesham Bucks

Distributed by S M Distribution
London SW9 01 274 8611 Telex 261643

Subscriptions
You can have *Popular Computing Weekly* sent to your home the subscription rate is £19.95 per year for addresses in the UK.

How to submit articles
Articles which are submitted for publication should not be more than 1000 words long

All submissions should be typed and a double space should be left between each line

Programs should whenever possible be computer printed

At present we cannot guarantee to return every submitted article so please keep a copy

Accuracy
Popular Computing Weekly cannot accept any responsibility for any errors in programs we publish although we will always try our best to make sure programs work

This Week



Cover [illegible] Craig

News 5
Spectrum add-ons Commodore software competition

Club Reports 7
Mick Ryan reports on the Independent Commodore Products Users Group

Labyrinth 8
Dave Middleton presents an amazing maze game for you to play

Reviews 10
Pinball Galaxy Invader disc drive for the Vic-20

Open Forum 13
Seven pages of programs

Sound & vision 20
Sam Blythe on music Brian Reffin Smith on art

Nand & mouth 21
John Gowne on calculators John Dawson on languages

Programming 23
BBC functions by Tim Hartnell

Peek & poke 25
Your questions answered

Competitions 26
Crossword puzzles a £10 prize

Editorial

After the initial enthusiastic reaction to the launch of the ZX Spectrum we are beginning to hear some of the first criticisms of the product

The Spectrum does not have a true moving-key keyboard

In fact its keyboard is similar in design to that of the ZX81 The keyboard consists of a flat sheet of 40 keys Over the top a moulded rubber sheet is suspended by plastic posts

Although the new design of the keyboard does allow you to type faster, it is still impossible to use it as fast as a conventional typewriter

It also looks as though Sinclair has used some clever tricks to maintain and refresh the screen display

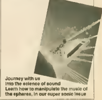
We still need to explore the extent to which this slows the computer down

There is obviously still a great deal to learn about the ZX Spectrum

Potential buyers, software authors and hardware add-on companies all need to learn as much about the machine as quickly as possible

Next week we start a regular page on the ZX Spectrum, covering aspects of its use and design

Next Week



Journey with us into the science of sound
Learn how to manipulate the music of the spheres, in our super sonic issue

ZX MICROFAIR

2 DAYS

New Century Hall, Corporation Street,

MANCHESTER

Saturday 29 May 1982 (10 30—8 30)
Sunday 30 May 1982 (10 30—5 00)

THE SHOW **EXCLUSIVELY** FOR THE SINCLAIR COMPUTERS NOW IN MANCHESTER

EVERYTHING FOR YOUR ZX80/81

SEE THE ZX SPECTRUM

Big selection from leading suppliers

COME ALONG AND SEE HARDWARE, SOFTWARE, BOOKS, PROGRAMS, MAGS, BRING AND BUY, USER GROUPS

TICKETS
(DOOR) ADULTS 50p CHILDREN 30p
(Under 10 Free with Adult)
(ADVANCE) ADULTS £1 CHILDREN 50p
Cheques to ZX Microfair

Organiser Mike Johnson 71 Park Lane Tottenham London N17 0HG

WE CHALLENGE YOU, EARTHLING.

Load this COSMOS program in your ZX81 16k and see if you can defend your despicable Earth convoys against our superior attack ships and warp mines

A totally new space game concept written in machine code to produce fiendishly clever effects and deadly results

You poor Earthlings have no chance

£5 99 ON CASSETTE.

PLUS OTHER 16K PROGRAMS ALSO ON CASSETTE

TRIAD *Three new puzzles to perplex and frustrate even the Rubik Cube expert. All with SAVE facility* ***£5 99***

QUARTET *An exciting compendium of crosswords puzzles and memory tests with up to 10 grades of difficulty* ***£3.99***

TABLE TUTOR *Using concepts within a schoolchild's experience this menu-driven tutor includes instruction table lists speed tests homework and answer service* ***£3 99***

PET LISTINGS
Black Box Mastermind Pontoon
£1 60 each or 3 for £4 00

Only high quality cassettes used
Send cheque or p o to

VORTEX SOFTWARE (DEPT POP)
26 CRAWFORD ROAD,
HATFIELD HERTFORDSHIRE, AL10 0PG.

Vortex SOFTWARE

MAIL ORDER ONLY PLEASE SEND SAE FOR CATALOGUE

TAURUS COMPUTER DESIGN

announce their new products for the [illegible] ZX81 [illegible]
A [illegible]
As seen at the Computer & Microfairs

TAURUS 8K EPROM UTILITIES MODULE

With EPROM you can forget the problems involved in loading the ZX81. Programs are instantly available at switch on

* Mounted neatly inside the computer. Fits memory space between 8K and 16K. Runs from computer power supply
* A type 2732 4K EPROM loaded with the [illegible] TAURUS machine code utilities package acclaimed by experts as the best available. Machine code MONITOR and a [illegible] to help you develop and debug [illegible] programs
  - Memory byte display and alter
  - Register display and alter
  - Word fill memory
  - Block copy
  - Generation of a REM statement
  - Tabulated memory display with addresses
  - Viewable user screen separate from monitor display screen
  - Hex to decimal and decimal to Hex converter
  - Routine to enable user to write his own routines in RAM
  - Facilities to run machine code and set breakpoints

Also in the package is the new TAURUS machine code DISASSEMBLER

* A type 2732 4K EPROM to give you the ability to develop and save your own permanent programs which can be instantly loaded
* Comprehensive User Manual **Price £30 00**

TAURUS MACHINE CODE MONITOR AND DISASSEMBLER
[illegible] available on cassette with user manual @ £7 75p and £6 50p respectively

TAURUS [illegible] 16K Ram Pack with switchable 2K EPROM loaded with [illegible] machine code [illegible] available for your own programs. **Price £54 95 or £45 95**

The ancient [illegible] game of NIM for 16K machines
[illegible] user interface [illegible] to beat the computer [illegible] automatically [illegible] ZX81 [illegible]
Cassette [illegible] user [illegible] £7 00

Send for details or cheque with order to
TAURUS COMPUTER DESIGN
47 High Street Baldock
Hertfordshire SG7 6BG
Telephone Baldock (0462) 893003

LASERBUG

Whether you are a BBC Microcomputer owner, have limited access or are merely interested then **LASERBUG** has something to offer you

LASERBUG is an independent users group dedicated to the support and enhancement of the BBC Microcomputer

LASERBUG is a central point for Ideas and Information — we want to hear from you and pass on your ideas and discoveries. Write to us at the address below — but please enclose a SAE if you want a reply

LASERBUG newsletter is distributed to members each month and is full of news, reviews, letters, articles, programs, contacts and classifieds (rates on application). Our regular meetings also go a long way to help you get the best out of your BBC Microcomputer

You want to join? for £12 00 you get a year's subscription to **LASERBUG**, 12 issues of the newsletter and free entry to all **LASERBUG**'s meetings and shows — or send £1 and a large SAE (12" × 9") for a sample copy of the **LASERBUG** newsletter and all the latest information

Write to

LASERBUG
4 Station Bridge, Woodgrange Road, Forest Gate, London E7 0NF

News



A Print 'n' Plotter graphic

Guide to graphics

Print 'n' Plotter Products have produced a 24 page guide to graphics programming on ZX81, for use in conjunction with their other micro drawing aids — the Print 'n' Plotter Jotter and Film. They are available from Print 'n' Plotter Products, 19 Borough High Street, London SE1 9SE. The guide costs £1.50 including postage.

The word is Oasis

Oasis Software offer two new word-processor packages for the ZX81: ZTEXT (upper case characters only) and LTEXT (both upper and lower case characters). LTEXT has been made possible by building a complete set of lower case characters using the high resolution graphics feature of the micro.

Both programs offer a text editor and a formatter/printer. The text editor offers string search and replace and merge facilities and the formatter/printer will produce justified text.

Oasis Software, Lower North Street, Cheddar, Somerset, is supplying both programs, together with detailed operating instructions, for £7.50.

UMIST's Microfest

UMIST (University of Manchester Institute of Science and Technology) is holding its first annual Microfest computer fair on 23 and 24 June. Micro clubs, user groups and local businesses will be represented as well as the main micro manufacturers.

Enquiries to Bancroft Hewitt Ltd, 121 Princess Street, Manchester M1 7AG, who will supply details.

Now Spectrum has add-ons too

RD Laboratories plan to be first with a Spectrum add-on.

Bob (RD) Dickens told *Popular Computing Weekly* that he is working on an adapter board for the Spectrum.

This will allow his RD 8100 range of realtime interface modules to be used.



The adapter board will be available in June, and will be followed by a new motherboard designed to be compatible with the Spectrum.

More details from RD Laboratories, 5 Kennedy Road, Dane End, Ware, Herts SG12 0LU.



Moving with the times ... The Mate from Database

Just check this out!

Good news for chess players — The Mate, a plug-in chess game from Applied Concepts (makers of Boris and Morphy) is now available for the Apple II, with a PET version due in August. It is not clear when a Vic version will be produced.

The Mate, with nine levels of play ranging up to a respectable USCF 1800, can make all moves, including en-passant and castling. It also has a special facility enabling it to monitor games between two human players.

The manual supplied with the game fully describes the interface software, making it possible to develop your own chess programs.

Check out The Mate at Database, 101 Cricklewood Broadway, London NW2 3JG.

Low turnout at ZX Microfair

Though fewer than 5,000 turned up for the ZX '82 Microfair — on April 30 and May 1 — organiser Mike Johnson said he was "not at all disappointed with the response".

Held just one week after the Earls Court Computer Fair, which had attracted some 36,000 people, the ZX fair at Westminster Central Hall featured well over 130 exhibitors.

Mike goes ahead with his Manchester ZX show, first reported in *Popular Computing Weekly*, at New Century Hall on 29-30 May. Over 60 stands are planned and he commented that the bookings were going very well.

Volunteer helpers are needed for the end of May show and should contact Mike Johnson at ZX Microfair, 71 Park Lane, London N18 0HG.

Program cash to be won

Commodore has announced a program-writing competition with prizes amounting to over £1,500.

Any program for a Vic 20 or PET is eligible, up to a maximum 32K RAM. The competition is open to individuals or group school entries and multiple submissions on cassette or disk are permissible.

The judges include Commodore's technical manager, a leading educational computer consultant, and Mike Todd, chairman of the Vic-Users Group.

The first prize comprises Vic single-drive floppy disk unit, Vic printer and Vic programmer's aid cartridge; the second and third prizes are a disk unit and a printer, and there are prizes for the runners-up.

For details of the competition and entry forms write to Commodore Software Competition, 35 Garway Road, London W2 4QD. The competition closes on June 30, 1982.

Two more magazines

Computer Games Review published bi-monthly from June will, as its name implies, look at commercially available games programs.

The Atom User will be for owners of Acorn micros, each month printing programs and reviews, and will include a question and answers section.

For details contact: Selwyn Ward, Computer Games Review, Computer Publications Ltd, 10 Star Lane, St Mary Cray, Kent, and, The Atom User, Acorn Computers Ltd, Fulborn Road, Cherry Hinton, Cambridge.

Drumming up trade

Trader Jack is the latest game for ZX81 from Luton software makers, Workforce.

The object is to deliver a variety of commodities by ship around an idyllic scattering of Pacific islands.

Trader Jack costs £7 including VAT and postage from Workforce of 140 Wilsden Avenue, Luton, Bedfordshire.

Club Reports

Is your club involved in any special projects? Use this page to tell the world about it.

Starting from scratch in the south-east

***Eli Jacobson** describes the founding and subsequent success of the SE ICPUG*

The Independent Commodore Products Users Group (ICPUG) recently renamed from the Independent Pet Users Group (IPUG) has a number of regional 'offsprings' Here we take a look at the history of one such user group — the South East Regional branch of ICPUG.

IPUG national sent Mick Ryan in Sevenoaks a list of IPUG member names and addresses from which he extracted 57 names living in Kent, Sussex and south of the Thames London Mick was already on the committee of the North Kent Amateur Computer Club (NKACC), which held its meetings at a school in Biggin Hill

Because the hire of the room was only £1 per night, and it was reasonably near to the centre of gravity of the 57 IPUG members, Biggin Hill was used for the first exploratory meeting, which took place in May 1980.

Thirty of the nearest members were telephoned, because this was the cheapest and quickest form of contact The meeting was an open evening for the NKACC, where Pet Users were showing their machines, and about 25 people attended

Swapping programs and ideas

Following this success, a more formal and separate evening was arranged for June, again based on a Pet program and ideas swapping evening. All 57 members were sent a circular and about 35 attended

Thanks to the co-operation of Helen Elsam from Commodore Mick was able to meet Jim Butterfield at the Cafe Royal Pet Show (those were the days!) and he kindly agreed to address the first proper meeting A good start!

There was no mention of subscriptions or organising committee yet, and costs were being kept to a minimum, all of which were borne by Mick on the promise of reasonable recompense from IPUG National

Jim Butterfield proved difficult to follow However the members' main interest seemed to be obtaining help with their programming problems Commodore seemed very helpful, so for the cost of the return fare, and a supper presented by Mick's long suffering wife, Commodore's resident software expert Paul Higginbottom readily agreed to run a programmers' clinic in July This was announced in the first formal newsletter.

Mick Ryan — man with a mission

Choosing committee members

Committee members needed to be reliable, willing, useful, and living close to Sevenoaks It was no use asking for votes at this early stage in the club's development 'military democracy had to be executed in 30 seconds before Paul began his clinic 'You, you and you' were volunteered by Mick for the committee, and agreed by the 40 or so members present before they could object!

Mick held the first committee meeting at his house in August It was a long affair, but covered all the necessary details for administering the group and planning the rest of the programme until the end of the year.

The IPUG constitution was adopted, and IPUG national kindly agreed to the use of their logo (with the addition of South East) to be used on the newsletter and letterheads

There were now 60 names on the mailing list Total set-up costs amounted to £25 and this was refunded by Commodore through IPUG national

The crucial decision was whether or not to have a newsletter Mailing was necessary to give members notice of meeting dates, venue and subject, so it was decided to add news items as they became available It would also have the plan of a permanent meeting locations and a list of officials The newsletter has attracted a great many members well outside the South East region

In November 1980 it was decided to join the Association of London Computer Clubs, but as it turned out IPUG South East was not to join the London Computer Fair until the third fair during Easter 1982

During 1981, the club went from strength to strength Membership was growing rapidly and by November 105 people had enrolled, and the group had about £700 in its bank account

The *Superscript* breakthrough

Towards the end of 1981 Simon Tranmer produced his superb new word processing package, known as *Superscript*, which has already allowed the group to buy a set of 8050 disk drives for use at Club nights, for *Superscript* production, and for Simon's development work The group now also has its own compiler available for compiling any member's private programs

The group recently took stands at the North London Polytechnic Computer Fair and the Earls Court Computer Fair, and are appearing in the future at the Commodore Computer Show amongst others They recently gave the first showing of the 40-column 64K Vic at a club night on the 29th of April The future can only see them grow and grow

That's how one group have done it How about you?

Write to Club Reports, *Popular Computing Weekly* Hobhouse Court, 19 Whitcomb Street, London WC2 7HF with details of successes you have had with your club with ideas for helping clubs along and with any news of special meetings We look forward to hearing from you

COVER STORY

Labyrinth

Learn how to elude the leery lout who lurks in the Labyrinth. A game by Dave Middleton

Slumbering in a complicated maze of tunnels is a fierce, fire-breathing dragon. As Knight of the Realm it is your duty to risk life and limb to enter the labyrinth of tunnels and kill the dragon.

In order to kill the dragon you have a bow and a quiver of five arrows. The arrows can be guided down as many as five tunnels, but if an arrow strikes a wall then it can bounce down any tunnel and may hit you.

Other dangers lurking in the maze are pits which you can fall down and gigantic bats which can pick you up and carry you off.

The dragon has BO so you can smell him when you are within one tunnel distance. You can also hear the flapping of the bats' wings and feel the draught from the pit, so you have no excuse for failing foul of any of the dangers within the maze.

The dragon will wake up whenever you fire an arrow or when you enter his room. He will then stagger to the next room but living in caves has done nothing for his eyesight and he may wander around in a circle and end up back at the same place. There is a one-in-four chance that he will stay where he is.

If he enters your cave then he will fry you with his flame thrower.

The program

The maze is designed around a dodecohedron, which is a 20-sided figure with three sides to each corner.

Because the ZX81 does not have DATA statements it is difficult to assign values quickly to an array without having a LET statement for each value, e.g., 10 LET S(1,5)=7.

To get round this a string of 60 characters was set up. By moving along the string and finding the CODE value for each letter and then subtracting 37, all the numbers between 0 and 20 can be generated. This is performed in lines 140 to 250.

Lines 250 to 350 set random positions for the hazards, yourself and the dragon. A check is made to ensure that all the positions are different.

Lines 380 to 630 is the main routine, calling the subroutines as required.

Subroutine 2000 tells you of any hazards, gives your position and the tunnel radiating from your cave.

Subroutine 2500 asks you if you want to move or shoot an arrow, setting the variable G accordingly.

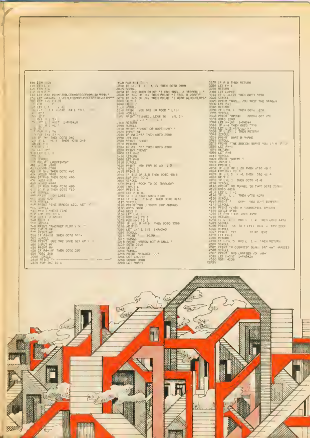
Subroutine 3000 shoots an arrow up to five caves and sets the variable F according to what happens, i.e., arrow hit dragon or yourself.

Subroutine 3500 moves the dragon and if he is in the same room as yourself burns you up.

Subroutine 4000 is the move routine. If you enter a room with a hazard then it is actioned immediately.

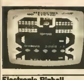


Illustration by James Macdonald



```
100 DIM
110 DIM L
120 DIM
130 DIM P
140 LET A$="ICHMTJODLCEHMOFEGOFHOA IH?P?K"
150 LET B$="I>CIMLNJSMOFIPOCIOPPIO>PIMP"
160 FOR I=1 TO 20
   [illegible]
   LET L
   LET     FLOOR  FOR L TO 1
   NEXT
   NEXT
   FOR I=1 TO
   LET L   INT  1+RND*20
   LET     L 1
   NEXT
   FOR     TO
   FOR I=1 TO
   IF     THEN GOTO 340
   IF L       THEN GOTO
   NEXT I
   NEXT I
   LET
   LET L
   
   SCROLL
400 PRINT "LABYRINTH"
   GOSUB 2000
   GOSUB 2500
   IF     THEN GOTO 460
   GOSUB 3000
   IF F=0 THEN GOTO 400
   GOSUB
   GOSUB 4000
   IF F=0 THEN GOTO 400
   IF F 0 THEN GOTO 10
   SCROLL
   PRINT "YOU LOSE"
   GOTO 520
   SCROLL
   PRINT "THE DRAGON WILL GET YOU
   SCROLL
   PRINT "NEXT TIME
   FOR I=1 TO
   LET L    
   NEXT I
   SCROLL
   PRINT "ANOTHER PLAY Y/N
   INPUT A$
   PRINT A$
   IF A$=N THEN GOTO
   SCROLL
   PRINT "USE THE SAME SET UP Y
   INPUT A$
   PRINT A$
   IF A$="N" THEN GOTO 200
   GOTO 360
   SCROLL
   PRINT "-------------------------------"
   FOR I=1 TO 6
2030 FOR
2040 IF     THEN GOTO 2080
2045 SCROLL
2050 IF J=2 THEN PRINT "I CAN SMELL A DRAGON"
2060 IF J=   OR J=4 THEN PRINT "I FEEL A DRAFT"
2070 IF J=5 OR J=6 THEN PRINT "I HEAR WING-FLAPS"
2080 NEXT I
2090 NEXT J
2100 SCROLL
2110 PRINT "YOU ARE IN ROOM " L(1)
   SCROLL
   PRINT "TUNNELS LEAD TO "
   
2160 RETURN
2500 SCROLL
2510 PRINT "SHOOT OR MOVE S/M? "
2520 INPUT A$
2540 IF A$<>"S" THEN GOTO 2580
2550 LET O=1
2560 PRINT "SHOOT
2570 RETURN
2580 IF A$ "M" THEN GOTO 2500
2590 PRINT "MOVE"
2600 LET O=2
2610 RETURN
3000 LET F=0
3010 SCROLL
3020 PRINT "HOW FAR TO GO 1-5
3030 INPUT D
3035 PRINT D
3040 IF D 1 OR D 5 THEN GOTO 4010
3050 FOR K=1 TO D
3060 SCROLL
3070 PRINT "ROOM TO GO THROUGH?
3080 INPUT L
3090 PRINT L
3095 LET P K =L
3100 IF K 3 THEN GOTO 3140
3110 IF P K    P K-2  THEN GOTO 3140
3115 SCROLL
3120 PRINT "NO U TURNS FOR ARROWS
3130 GOTO 3000
3140 NEXT K
3150 LET L=L 1
3160 FOR K=1 TO D
3170 FOR M=1 TO 3
3180 IF   L M =P K  THEN GOTO 3500
3190 NEXT M
3200 LET L=  L INT  1+RND*3
3206 SCROLL
3210 PRINT "   DOING
3214 SCROLL
3215 PRINT "ARROW HIT A WALL "
3220 GOTO 3310
3230 NEXT K
3240 SCROLL
3245 PRINT "MISSED "
3246 LET L=L(1)
3250 GOSUB 3500
3260 LET
3270 IF A B THEN RETURN
3280 LET F= 1
3290 RETURN
3300 LET L=P(K)
3310 IF L  L(2) THEN GOTO 3050
3320 SCROLL
3325 PRINT "AHA    YOU SHOT THE DRAGON
3330 LET F=1
3340 RETURN
3350 IF L L 1  THEN GOTO
3355 SCROLL
3360 PRINT "ARGHH    ARROW GOT YOU
3370 GOTO 3280
3500 LET K=INT  1+RND*4
3510 IF K=4 THEN GOTO
3520 LET L 2 =S(L(2) K)
3530 IF L 2   L THEN RETURN
3540 SCROLL
3550 PRINT "WHAT A SHAME
3560 SCROLL
3570 PRINT "THE DRAGON BURNT YOU TO A
3580 LET F=-1
3590 RETURN
4000 LET F=0
4010 SCROLL
4020 PRINT "WHERE T
4030 INPUT L
4040 PRINT L
4050 IF L 0 OR L 20 THEN GOTO 40
4060 FOR K=1 TO 3
4070 IF   L 1   K =L THEN GOTO 41
4080 NEXT K
4090 IF L=L 1  THEN GOTO 41
4100 SCROLL
4110 PRINT "NO TUNNEL IN THAT DIRECTION
4120 GOTO 4010
4130 LET L 1 =L
4140 IF L  L    THEN GOTO 4250
4150 SCROLL
4160 PRINT "     OOPS  YOU JUST BUMPED
4170 SCROLL
4180 PRINT "INTO A SLUMBERING DRAGON
4190 GOSUB 3500
4200 IF F=0 THEN GOTO 4490
4210 RETURN
4250 IF L  L 3  AND L  L 4  THEN GOTO 4490
4255 SCROLL
4260 PRINT "YOU JUST FELL INTO A DEEP
4265 SCROLL
4267 PRINT "PIT           THE END
4270 LET F=-1
4280 RETURN
4490 IF L L 5  AND L  L    THEN RETURN
4500 SCROLL
4505 PRINT "A GIGANTIC BLACK BAT HAS GRABBED
4506 SCROLL
4507 PRINT "AND CARRIED YOU AWAY
4510 LET L=INT  1+RND*20
4520 GOTO 4130
```

Reviews
software



Electronic Pinball

Available from any Commodore Vic dealer Price £19.95

This Commodore product comes in the form of a plug-in cartridge, and connects straight up to the Vic

On power up, the instructions for playing the game are displayed on the screen rather off-centre Don t adjust the centring just yet though read the instructions first (you can see most of them), and when the screen shows the pinball table centre it up then.

Pressing function key 1 gets the ball rolling, and then you press function key 1 or 3 depending on whether you want a one or two player game respectively This game may hit your wallet, however, because you require the Vic paddle to play it

The game itself is very exciting, and the use of the Vic s high resolution graphics, colour and sound is excellent Once the ball is released, you have two blocks with which to stop the ball disappearing down the centre of the table

At the top of the screen is a wall of small coloured blocks, which you have to knock out in 'breakout fashion If you manage to do this, a strange alien being appears in its place, and if you kill the 'being' (just by hitting it) a random bonus score is achieved

In the top centre of the screen various other aliens appear from time to time and bonus points are also awarded for hitting those

One of the very interesting features of this game is a row of 'faces towards the bottom of the screen As the ball passes over them they turn from a frown to a smile, and making all five of them beam happily away gives you yet more bonus points

This is not as easy as it sounds, because if the ball passes over a particular face again, it reverts from a smile to a frown, so to get all five smiling at the same time is hard

A bonus ball is achieved if you reach a score of 50 000 which is not often done I believe the highest score recorded at the recent Hanover Fair, for instance, was just over 35,000

Summary

A very good and very addictive game, which makes full use of the programming facilities available on the Vic Seeing games like this makes you wish that more writers of games software for the Vic would adhere to this high standard **PG**

The Bible

Automata, 65a Osborne Road, Portsmouth, Hants
ZX81 1K, cassette, price £5 00

Surely one can get Automata under the Trade Descriptions Act? 'First Edition is the subtitle of this third cassette of its 'adult games' for the 1K ZX81 But I'm sure I've come across the name before — there s a book of the same name, I think

The Bible costs more than two earlier cassettes Perhaps this is to ensure that only adults (and not clergymen) buy it However, we now get 10 programs instead of eight, and the quality has improved significantly

'Quality' in more than one sense For a start, the programming is better — the screen is used more effectively, and graphics are generally impressive (for 1K)

The other kind of quality concerns adultness The two earlier cassettes tended to be so crude (sex-wise) that even broad-minded viewers found them intolerable

The Bible is still for adults, but it is humorous rather than crude and only a bit bawdy in places

The 10 programs themselves are nearly all addictive They deal with such matters as 'Adam and Eve' (a battle between God + angels and the Tree + devils to get Adam s soul), 'Plagues (get each one to land on Pharaoh s head as he dodges to and fro), and 'Jonah (your task, as the whale, is to swallow J before the sea serpent gets him)

Automata has polished its audio commentary too (it s been a unique and effective feature of each cassette) With an excellent range of relevant sound effects, we hear such bits as 'and God said, "Noah, get thou thy welllie boots" '

I hope all these forms of improvement continue, and look forward to Automata s next 1K cassette

Summary

Ten interactive 1K games, almost all good and well-documented, plus neat audio 'atmospherics' **KJ**

Galaxy Invaders

Bridge Software 36 Fernwood Marple Bridge, Stockport, Cheshire
ZX81 16K, cassette, price £3 00

There are a good number of ZX81 invaders available Bridge Software s version has been around a while now It is one of the two best (the other is that from Macronics) and is deservedly popular

As far as screen display and program usage are concerned, Bridge Software s Invaders is quite standard RUBOUT to fire, 5 and 8 to move, five rows of oscillating shimmering, advancing 'vaders, three shields You can move while firing and should find no trouble in getting your fingers habituated to hours of action

After loading the cassette (with some difficulty if my experience is typical), you have a choice from 10 levels of play Even the lowest levels do no good for one s adrenalin production, at the highest level it takes no more than five seconds (yes, five seconds) for the bottom row of invaders to reach the top of the shields

When this happens, or when you re zapped, the game restarts at once Four sets of numbers are displayed continuously — highest score, last score, current score (and, for the less able, number of shields remaining, rather than lowest score)

The BREAK interrupt is masked while this program is running However SHIFT will stop the game, clear the scores, and return you to the start

Summary

An excellent version of a standard arcade game Good value

Reviews
hardware



VIC-1540 disk drive

Commodore Business Machines, 675 Ajax Avenue, Slough Trading Estate, Slough, Buckinghamshire
Price £396

The Vic disc drive is priced at £396 including VAT, which could be said to be too expensive. Indeed, at twice the price of the basic Vic there is no other real way of looking at it. So, what are you getting for your money?

The basic unit: the disc drive is conveniently packaged in a fairly smart box, and comes complete with manual, test disc, connecting cable from the Vic to the drive, and a mains lead (but no plug on the end? Surely manufacturers can afford plugs! At £396 they ought to be able to).

With the disc unit comes a sample disc containing seven programs (the listings are displayed in the manual if the disc fails to operate, although mine worked quite happily). These are a form of DOS support, to simplify the use of the disc in immediate mode, but not in program mode.

There are a number of programs to check out the performance of your unit and look at information stored on the disc together with demonstration sequential and random access programs.

The disc drive: the Vic disc drive has a very smart and pleasing appearance, it stands just 10 cms high by 20 cms wide. Taking standard 5¼ in diskettes, it has a capacity of just over 174K, stored in the form of 664 blocks of information.

It is read/write compatible with the existing Commodore 4040, and read-only compatible with earlier 3040 and 2040 discs. Dics formatted on an 8050 haven't got a chance!

Files can be stored either as program, sequential or relative files, although relative file handling is rather difficult to perform, as the Vic itself does not have the DOPEN commands and so on.

Sequential file handling is (if you'll excuse the pun) relatively easy to operate, and goes a long way towards turning the Vic into a business computer.

How many programs can you store on a disc? That will obviously depend on the size of each individual program, but if you're using a standard Vic that would mean no program is going to be more than 3.5K long; as the Vic disc drive holds 174K, this gives you the option to store around 50 programs per disc.

Summary

The Vic disc drive is a welcome addition to the Vic range of peripherals. Its neat, compact design fits well into the Vic family tree, and the ability to very quickly LOAD and SAVE programs and files is almost, a necessity. I've yet to discover any 'serious' bugs in it: one or two idiosyncracies, perhaps, but those are only to be expected.

The documentation could be better, but it is certainly adequate from a beginner's point of view. The read/write compatibility with the earlier PET 4040 disc drives is extremely useful, and the ability to read discs formatted on 3040/2040/2031 disc drives is similarly useful.

Of my two main complaints, one is inherent in it being a single disc drive: you can't, other than tediously, make a backup copy of a disc! My biggest complaint is price: at £396 *is* expensive, and many people may think twice before purchasing.

It probably will not be long before some cheap 2¼ in disc drives come on to the market. It could be worth waiting for them. **PG**

ZX81 input-output port

Bolton Electronics, 44 Newland Drive, Bolton, Lancashire
Price £15.90 ready-made, or £12.90 kit. Postage and packing £1.00

This single printed circuit board will provide two connections to the outside world, one for input and one for output.

Each port consists of eight bits or wires which can be used to communicate with devices outside the ZX81. The port can be built from a kit of parts which consists of four standard TTL logic chips, diodes, capacitors and two 16-pin IC sockets (for connecting up to external devices).

These sockets also provide connections to the +5 volt and 0 volt lines, essential if equipment is to be driven from the port. The instructions for building the kit are very poor, two bad photocopies of the board showing the components and an incomplete circuit diagram. The electrolitic capacitor is also shown the wrong way round.

When building the kit it is very easy to connect up two of the tracks. The board has no plating through the holes, so numerous connections have to be made with wire from one side to the other.

The User Instructions that come with both the kit and the ready-built port version are completely the opposite to the kit instructions. Apart from a last-minute change of address from 65535 (which clashes with the tv screen) to 9999, the instructions are very clear and easy to understand.

Binary notation is explained, and several programs to demonstrate the use of the port make the learning easy. The instructions contain hints on what can be done with the port, such as controlling relays (a 240 volt, 5 amp version which can be driven from the port costs £2.59 from the same firm) or measuring the temperature via the input port by using the simple one chip circuit supplied.

The connection to the ZX81 is via the usual edge connector, but the connection to the printer or 16K RAM pack is made on the bottom edge of the vertical pcb. This raises the back of the computer by between 30° and 45°, but the RAM pack is held firm on the table or whatever. The keyboard is also now at the correct angle for typing.

Summary

The ready-made port is cheap and easy to use, but add-ons may cause problems unless the user can solder a pcb onto the back of the port. The kit can cause more problems than the £3 saved, and is best to pay the extra. **SA**

Open Forum

Open Forum is for you to publish your programs and ideas
It is important that your programs are bug free before you send them in We cannot test all of them
Contributions should be sent to: Popular Computing Weekly, Hobhouse Court,
19 Whitcomb Street, London WC2H 7HF.

Snake in the grass

on BBC Micro

So many games routines involve something bouncing around within a box that this was one of the first things worth investigating with the BBC Microcomputer

These routines make a 'snake (a short line) move in this way, setting up the routines is far more interesting than the result, and should teach you a great deal about programming

First we set up the colour definitions

```
10 MODE 5
20 GCOL 0 130              (background to be yellow)
30 VDU5                    (switch off cursor)
40 CLG                     (present that background)
50 GCOL 0 0                (foreground to be black)
```

The first exercise involves drawing a line

```
60 X = 30 Y = 35           (define start-point)
70 MOVE X Y                (go there)
80 XI = 6 YI = 6
        (define increments — try changing them)
90 X = X + XI Y = Y + YI   (step along line   )
100 PLOT 69 X Y (    plotting foreground point    )
130 GOTO 90                (    at each step)
```

Then we introduce the 'bounce' idea This involves reversing the increment sign each time the line reaches an edge of the box

```
110 IF X>1250 OR X<30 THEN XI = -XI
120 IF Y>1000 OR Y<25 THEN YI = -YI
```

The result is an utterly boring yet utterly fascinating routine in its own right One can spend hours with it, trying such variations as these

○ Different colours (lines 2 and 5 and use of the VDU statement)
○ Different, and differing, increments (line 8)
○ Different boundaries (lines 11 and 12)

No two patterns are alike, many are remarkably hypnotic However, games involve bouncing balls rather than spaghetti junctions of lines, so one must erase the trail behind the head of the line

```
55 A = 1
100 PLOT 71 X - XI * A Y - YI * A
```

PLOT 71 means plot a point at the co-ordinates marked, in the background colour

You should now be able to develop a BBC *Breakout* But we still have my snake to think about Increasing the value of A (line 45) provides one It also provides the problem of grass growing under the snake's feet

You may just as well use that, to obtain the last hypnotic routine Keep line 55 as it is and edit lines 110 and 120 by inserting A=A + 1 at the end of each.

Generating pretty patterns is just as much fun as writing games To get rid of the grass is another story It involves setting up a separate set of co-ordinates and increments for the PLOT 71 to operate on Try it, add some interaction and there's a race game for you!

YOUR PROGRAM COULD WIN A PRIZE!

Each week the editor goes through all the programs that you send to Open Forum in order to find the Program of the Week.
The author of that program will qualify for DOUBLE the usual fee we pay for published programs (the usual fee is £10).
Then at the end of the month the four best programs of the week go forward to our amazing Program of the Month contest, for which there is a STAR prize
This month the star prize is a super ZX printer, worth £59 95!
And at the end of the year, all the best Programs of the Month will be entered in the super colossal competition, Program of the Year.
So send in your program today!

Programs which are most likely to be considered for the Star Prize will be computer printed and accompanied by a cassette
The programs will be well documented the documentation being typed with a double spacing between each line
The documentation should start with a general description of the program and then give some detail of how the program has been constructed and of its special features
Listings taken from a ZX Printer should be cut into convenient lengths and stuck down on to white paper
Please enclose a self-addressed envelope

Horse racing

on ZX81

Spend a day at the races without leaving your own home

The program is written for 4K memory and above, and is a simulation of a horse race The horses are presented along with the starting prices, you are then invited to select your horse and place a bet which may be either 'on the nose' or each way

The race is then displayed showing the horses moving down the field When the first five horses have passed the post the results are displayed, showing the position of your horse along with your winnings and total cash

There is always a chance that having won the race an objection is lodged and your horse is disqualified, possibly on the ground of only having three legs, who knows?

Listing 1 shows the Basic program Lines 10 to 60 initialise the variables Lines 70 to 215 accept the inputs and checks their validity Lines 219 to 330 perform the actual racing Lines 1000 to 1120 make up the routine that calculates the odds Lines 2000 to 2295 save the first five winners, present the results and display your winnings

Lines 3000 to 3100 make up the routine dealing with any possible objection

To calculate the odds, a random number is generated and held in

continued overleaf

continued from page 13

SEED The random number generator is then started from SEED, the first 36 random numbers are sampled and the results placed in P() The numbers held in P() are then divided into 36 to give a ratio These are the odds

When the race is started the random number generator is restarted from SEED and the random numbers are selected and the respective horse is moved forward one, if that horse has not previously finished

After running the program it has been found that 36 is the optimum sample size The greater the sample the more likely the odds tend to even out at around 9-1

If the sample was less than 36 the odds would not give a fair indication As it is the odds if given at 4-1 do not necessarily mean that a horse will win, but it stands a good chance

The payout is calculated by dividing the finishing place into the odds so 1st pays 100%, if backed each way, 2nd pays 50%, 3rd pays 33% and 4th pays 25% plus your stake

Here's hoping you don't get too addicted to the game, but you can become obsessive in trying to win back your money

Glossary

T:	Total cash at hand
G:	No of races run, acts as a flag, if less than 10 when all your cash has gone gives a bonus of £10
HS:	String holding the horses numbers
W:	Counter to count the number of horses past the post
H():	Column position of the horses
P():	Starting prices of the horses
W():	Horses past the post
SEED:	Random number seed
H	Selected horse
B	Bet placed
TS	Type of bet — to win or each way
X:	General variable
R:	Horse presently being moved
FP:	Finishing place of selected horse
O:	Objection Flag 0=1 objection upheld, 0=0 objection rejected

Horse racing
By John Sylvester

```
1 REM ** HORSE RACING **
2 REM (C) SYLVESTER 1992
10 LET T=50
11 LET G=0
15 LET H$="123456789"
20 LET W=1
30 DIM H(9)
40 DIM P(9)
50 DIM W(5)
55 RAND
60 LET SEED=INT (65535*RND)+1
70 GOSUB 1000
75 PRINT AT 19,0,"CASH TOTAL=$",T
80 PRINT AT 21,0,"PICK A HORSE (1-9)"
90 INPUT H
100 IF H<1 OR H>9 OR H-INT H<>0 THEN GOTO 90
102 LET H$(H)=CHR$ (CODE H$(H)+128)
110 PRINT AT 21,0,"ENTER STAKE, HORSE ";H
    ;" SP=",P(H),"-1  "
120 INPUT B
130 IF B<=T THEN GOTO 160
140 PRINT AT 20,0,"YOU HAVE NOT ENOUGH MONEY"
150 GOTO 110
160 PRINT AT 20,0,"                         "
180 PRINT AT 21,0,
    "TO WIN (W) OR EACH WAY (E) ?"
190 INPUT T$
200 IF T$="W" OR T$="E" THEN GOTO 215
210 GOTO 190
215 IF T$="E" AND (2*B)>T THEN GOTO 140
219 LET T=T-(B+B*(T$="E"))
220 CLS
230 FOR X=1 TO 10
240 PRINT "                               "
250 IF X<=9 THEN PRINT H$(X)
260 NEXT X
265 RAND SEED
270 LET R=INT (9*RND)+1
280 IF H(R)>31 THEN GOTO 270
290 PRINT AT (2*R)-1,H(R)," "
300 LET H(R)=H(R)+1
310 IF H(R)=32 THEN GOTO 2000
320 IF H(R)<=31 THEN PRINT AT
    (2*R)-1,H(R);H$(R)
330 GOTO 270
1000 RAND SEED
1002 LET G=G+1
1005 CLS
1006 PRINT "STARTING PRICES -"
1020 FOR X=1 TO 36
1030 LET R=INT (9*RND)+1
1040 LET P(R)=P(R)+1
1050 FOR C=2 TO 9*H
1060 NEXT X
1065 SLOW
1066 PRINT ,,"HORSE" TAB 13,"PRICE",,
1070 FOR X=1 TO 9
```

Horse racing (cont'd)

```
1075 IF P(X)=0 THEN LET P(X)=.25
1080 LET P(X)=INT (36/P(X))
1090 PRINT TAB 2,X,TAB (13+P(X)<10)-(P(X)=100)),
     P(X) "-1"
1100 NEXT X
1120 RETURN
2000 LET W(W)=R
2010 LET W=W+1
2020 IF W<=5 THEN GOTO 320
2030 CLS
2031 LET O=0
2035 GOSUB 3000
2040 LET FF=0
2050 PRINT , "RESULTS",,,,
     "POSITION     HORSE      SP"
2060 FOR X=1+O TO 4+O
2070 PRINT TAB 4,X-O,TAB 14 W(X) TAB 22,P(W(X)) "-1"
2080 IF H=W(X) THEN LET FP=X-O
2090 NEXT X
2100 IF FP>0 THEN GOTO 2200
2110 PRINT ,,"HORSE ",H," NOT PLACED"
2120 PRINT ,,"WINNINGS=$" FP " STAKE=$" B
2124 IF T$="W" THEN PRINT " WIN"
2126 IF T$="E" THEN PRINT "E.W."
2130 LET T=T+FF
2140 IF T=0 THEN GOTO 2250
2150 PRINT   "CASH TOTAL=$" T
2160 PRINT ,,"PRESS S TO START NEXT RACE"
2165 LET H$(H)=CHR$ (CODE H$(H)-128)
2170 IF INKEY$="" THEN GOTO 2170
2180 IF INKEY$="S" THEN GOTO 15
2190 GOTO 2170
2200 PRINT   "HORSE " H," PLACED ",FP
2210 IF FP>1 AND T$<>"E" THEN LET FP=0
2220 IF FP=0 THEN GOTO 2120
2230 LET FF=B+B*(P(H)/FP)
2240 GOTO 2120
2250 PRINT , "NO MORE CASH"
2260 IF G<10 THEN PRINT "I'LL BUY YOUR SHIRT,
      HERES A     $10 NOTE"
2270 IF G<10 THEN LET T=10
2280 IF G<10 THEN GOTO 2290
2295 STOP
2290 LET G=10
2295 GOTO 2150
3000 IF RND>=.2 THEN RETURN
3005 PRINT "OBJECTION TO WINNER RAISED"
3010 LET R=RND
3020 IF R<.5 THEN LET O=0
3030 IF R>=.5 THEN LET O=1
3040 FOR X=1 TO 100
3050 NEXT X
3060 PRINT ,,"OBJECTION "
3070 IF O=1 THEN PRINT "UPHELD"
3080 IF O=0 THEN PRINT "REJECTED"
3090 IF O=1 THEN PRINT   "HORSE ",W(1),
      " DISQUALIFIED"
3100 RETURN
```

Pairs card game

on ZX81

This program allows the well-known card memory game of Pairs or Pelmanism to be played on an unexpanded ZX81. The cards, 54 of them (including two jokers), are laid out in a block 9×6 with numbered columns and rows.

The player inputs four numbers which represent the row and column of two cards. These cards are then displayed in their positions.

If the cards constitute a pair they are removed from play; if not they are turned over again after a short pause. The number of tries which the player has had so far is displayed underneath the block of cards on the screen.

The program is a typical example of an idea being made to fit the unexpanded ZX81 by brute force. The original version of the program took less than an hour to write. This final version, which just scrales in, took three times longer than that.

The process of shortening the program has led to the removal of almost all literal numbers and their replacement by variables: the use of logical statements to produce values (B≠B in line 110 producing zero) and multiple print statements (as at lines 340 and 350).

The most important variable in the program, A$, which contains the pack of cards, is nowhere declared in the program but entered in direct mode to save a considerable amount of space. A$ is, in fact, 123456789TJQK repeated four times (the four suits) plus ** representing the two jokers.

The program does not recognise absurd moves, nor does it recognise the end of the game when it comes, but since all the cards will have been removed it's difficult for a player to miss!

Program notes

Line 110: This is an example of space saving. B≠B takes three bytes in the program file — 0 would take 7. Having set 0, it is used throughout the program to produce other values where possible thus avoiding the need to define another variable.

Line 130: This loop shuffles A$.

continued overleaf

continued from page 15

Line 210: The graphics strings here consist of alternating graphics H and Space in 2 sets of 9, one starting with the Space and the second with the H
Line 350: This line illustrates not only how PRINT AT statements can be combined in a program line but also how any PRINT statement can be associated with a logical condition If IF . THEN had been used the statements would have required separate lines

Paire card game
by David Lawrence

```
1   LET  -
11  LE      E
12  LET A$  $
1   FOR I=A TO LEN A$
140 LET S=INT (RND*LEN A$+A
150 LET T$=A$ I)
160 LET A$(I =A$(S
170 LET A$ S =T$
180 NEXT I
190 PRINT " 123456789"
200 FOR I=A TO   TEP A+A
210 PRINT I '########   I+    ########"
220 NEXT I
230 INPUT E
240 INPUT F
250 INPUT G
260 INPUT H
270 LET C=C+A
280 PRINT AT $ $( $ C
290 LET   -(E A *$+F
300 LET Y= G-A *$+H
310 PRINT AT E F A$ X) AT G H A$ Y
320 FOR I=A TO $*$
330 NEXT I
340 PRINT AT E F  #" AT G,H '#
350 PRINT AT E F  #' AND //.( INT  X 2  AT G H  #  AND    _  INT
360 IF A$( )=A$(Y  THEN PRINT AT E "     AT G H
370 GOTO 230
```

Great circle

on ZX81

Travelling by air or communicating by radio, the shortest distance between two points on the earth's surface is part of a great circle of the globe

This program calculates the distance in degrees and then converts into nautical miles and finally into statute miles

For Bethlehem the first input would be 31 42 and the second −35 12 (because East), the program making the necessary conversion of the minutes to a decimal of a degree, eg, 31 7 and −35 2

In the examples the starting point was Bolton in Lancashire Remember that South latitudes and East longitudes require the minus sign

Great circle
by William Cartwright

```
0 REM 'GREAT CIRCLE DISTANCE'
1 REM WXYZ= LAT 1 LAT 2 LONG 1 LONG 2
2 PRINT "TO CALCULATE THE GREAT CIRCLE    DISTANCE BETWEEN TWO PLACE.
3 PRINT GIVEN LAT 1  LAT 2  LONG 1  LONG 2   SOUTH LATS AND EAST LONG ARE
  NEGATIVE'
4 PRINT "INPUT LAT 1
5 INPUT W
6 LET F=W
7 GOSUB 1000
10 LET A=S*PI/180
14 PRINT "INPUT LAT 2'
15 INPUT Y
16 LET F=Y
17 GOSUB 1000
20 LET B=S*PI/180
22 PRINT  INPUT LONG 1"
23 INPUT X
24 LET F=X
25 GOSUB 1000
26 LET L=S
27 PRINT  INPUT LONG 2'
28 INPUT Z
29 LET F=Z
30 GOSUB 1000
31 LET U=S
32 LET E=U-L
34 PRINT E
35 LET C=E*PI 180
40 LET D=(SIN A*SIN B)+(COS A*COS B *COS C
50 LET E=ACS D*180/PI
55 LET G=E*60*1.15
56 PRINT
60 PRINT  DISTANCE -    G      MILES
70 STOP
1000 LET T=ABS F
1001 LET M=T-INT T
1010 LET N=M*100
1020 LET P=N/60
1025 LET Q=N-INT N
1045 LET R=INT Q/36
1060 LET S=INT T+P+R
1061 IF T<>F THEN LET S=-S
1062 PRINT S
1063 PRINT
1070 RETURN
```

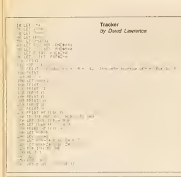
Tracker

on ZX81

From this simple program, which does little more than set up a board and add regular increments to two coordinates, comes an infuriatingly difficult game called *Tracker*

It's a hunting game, but the quarry is on the move For each hunt the quarry's move is set by lines 60 and 70, and its initial position by lines 80 and 90 A 12*12 board is displayed by the loop at line 100 and the player is invited to specify a square by lines 170 to 220

Once a square is chosen, the program marks it with an '0 and then indicates the direction of the quarry by placing a + in one of the adjacent squares Then the quarry makes its move Each move can be up to four squares on both axes and, as men-



Tracker
by David Lawrence

tioned, the move is constant during a given hunt

To have a real chance to catch the quarry quickly you must find a strategy which allows you to discover the direction in which it is travelling on both axes and the distance (always remembering that if it leaves one side of the board it reappears on the other)

That may sound easy but on a small board, with relatively large moves, it is quite possible to labour under the illusion that the quarry is travelling right to left when in fact it is going in the opposite direction!

Except to explain that the prompts 'D and 'A' refer to row and column on the board, that is all the help you get

Good Luck Sometimes you will need it!

Program notes

Line 10: Note how, to save memory the majority of values are based on logical manipulation of one literal number — it is literal numbers that eat up memory

Line 190: Why a single 'PRINT'? If you think about it, by far the easiest way to set a board and act upon it in 1K is to have the co-ordinates of the board beginning at 1 In this way players moves don't have to be translated (e.g LET M=M−1) every time thus saving program lines

Line 250: Here and at lines 260, 300 and 310 logical operators are used to achieve the effect of two IF THEN statements on a single line P+Z*(P<W) in line 300 simply means that 12 is added to P if P< W i e (P<W) is true and therefore equal to 1 rather than 0.

Knockout
by Lee Tanner

```
10 POKE 36879 [illegible] PRINT [illegible] KNOCKOUT
20 SP [illegible] 3(A*6 M [illegible] U+2
30 PRINT [illegible]
40 GET G$ IF G$< 1 OR G$ [illegible] THEN 40
50 IF G$="1" THEN B$ [illegible] wht shift U x 3}(space)" M=16
60 IF G$="2" THEN B$ [illegible] wht shift U x 2}(space)" M=17
70 IF G$="3" THEN B$ [illegible] wht shift U (space)" M=18
80 POKE 36879 [illegible]
100 PRINT"(cls" [illegible] POKE SP+J+I x 22 207 POKE CC+J+I x 22 1 NEXTJ I
110 BL=5 BT 11 POKE [illegible]
120 A=1 [illegible] 8+(I AND BL) [illegible] BL
130 FOR I 1 TO 10 POKE V 15 FOR [illegible] TO 10 NEXT J POKE V 0 FOR [illegible] 1 TO 10 NEXT J NEXT I
140 IF BL 0 THEN 300
150 POKE SP+A+B x 22 32
160 A=A+X B=B+Y
170 IF A=19 OR A=1 THEN X -X POKE V 10
180 IF B 0 OR PEEK(SP+A+B x 22)=207 OR PEEK(SP+A+B x 22 + 22) 120 THEN POKE SP+A+B x 22 32 Y -Y POKE V 15
190 IF B>20 THEN 120
200 POKE SP+A+B x 22 81 POKE V 0
210 IF PEEK(197) 29 THEN BT BT-1
220 IF BT<0 THEN BT 0
230 IF PEEK 197) 3 THEN BT BT+1
240 IF BT>M THEN BT M
250 PRINT"(home)(cd x 21 TAB BT)B$ GOTO 150
300 FOR I=1 TO 5 PRINT (home)(cd)(cr x 6)GAME OVER" FOR J 1 TO 200 NEXT J
310 PRINT (home)(cd (cr x [illegible] space x 9)" FOR J 1 to 200 NEXT J NEXT I
320 [illegible] FOR I=2 TO 6 FOR J=1 TO 19
330 IF PEEK(SP+J+I x 22)=32 THEN SC=SC+(7-I)
340 NEXT J I
350 PRINT" home (cd x 8)YOU SCORED"SC IF SC=285 THEN PRINT"(rvs)MAXIMUM"
360 FOR I=1 TO 5000 NEXT
370 PRINT [illegible] cd x 4)(cr x 5)PLAY AGAIN /" PRINT (cd x 2)( (pur)Y(wht) OR (pur) N POKE 198 0
380 GET G$
390 IF G$ N THEN POKE 36879 27 PRINT (cls)(blu)" POKE 50 0 END
400 IF G$ Y THEN 30
410 GOTO 380
```

Knockout

on Vic-20

Here is a program for the standard Vic 20

It is a version of the old arcade game *Breakout* in which you have to knock out the coloured bricks at the top of the screen using a bat and ball

There are three levels of play and four balls per game Level 1 is the easiest and level 3 the hardest

The bat is controlled by the < and > keys for left and right respectively

At the end of the game you are given a score The maximum score is 285

Music maker

on BBC Micro

This program allows you to enter your melody as a string, which the BBC Microcomputer then interprets and plays

The principle of the program is very simple The BBC Micro's sound command has four parameters, and is written in the form SOUND 1, −15, 128 4

The first number after the word sound chooses the channel (0 to 3), the second is the volume (−1 to −15, with −15 the loudest), the third number is the pitch (0 to 254) and the fourth is the duration (from one upwards).

The channel (parameter one) is fixed in this program to be channel one (the first number after the word SOUND) The volume varies randomly from −11 to −15, the pitch and duration are set by the melody which you enter as a string

Lines 40 and 50 set the initial display to tell you to 'Enter your song', and — once you have done this — to determine the speed, from 1 (very fast) to 9 (slow) The speed is accepted in lines 80 and 90

The main REPEAT/UNTIL loop, which actually turns the elements of the string into 'music', runs from line 110 to 200 Line 120 calls up a procedure (PROCdisplay) to print the words Music Maker in a random colour, on a randomly coloured background, on the screen

How to enter a song

The program works by accepting the notes you need as letters, running as follows CDEFGABcdefgabx Note that the highest 'c' is accepted as an x After each letter comes a number, which determines the duration of the note A rest is shown by a P (for 'pause') so a string which read A3B4c5P2c3 would play the note A for a count of three, followed by B for a count of four c for five, a rest of two then c again for three

The program will play the music over and over again until you press BREAK

There are three sample songs, which you can enter by typing MODE 4 RETURN, then GOTO 270 The program will quickly stop with an error

Music maker

By Tim Hartnell

```
10 REM *Music maker*
20 MODE 4
30 REM (C) HARTNELL 1982
35 REM FROM "Let your BBC Micro
37 REM      teach you to program"
40 VDU 19,3,3,0,0,0
50 VDU 19,0,4,0,0,0
60 PRINT'''
70 INPUT"Enter your song        "A$
80 PRINT"How fast? 1 (very fast) to 9 (slow)
90 SPEED$=GET$:TEMPO=ASC(SPEED$)-48
100 REM***********
110 REPEAT
120 PROCdisplay
130 FOR J=1 TO LEN(A$)-1
140 B$=MID$(A$,J,1)
150 N=-53*(B$="C")-61*(B$="D")-69*(B$="E")
    -73*(B$="F")-81*(B$="G")-89*(B$="A")
    -97*(B$="B")-101*(B$="c")-109*(B$="d")
    -117*(B$="e")-121*(B$="f")-129*(B$="g")
    -137*(B$="a")-145*(B$="b")-149*(B$="x")
160 D=VAL(MID$(A$,J+1,1))
170 IF B$="P" THEN GOTO 220
180 SOUND 1,-(RND(5))-10,N,D*TEMPO
190 NEXT J
200 UNTIL FALSE
210 REM**************
220 FOR Z=0 TO D*TEMPO
230 SOUND 1,0,0,0
240 NEXT
250 GOTO 190
260 REM ****Sample songs follow****
270 Z$="c1c1A2B1G1c1C1A2B1G1c1c1A2B1G1F1D5
    B1B1B2A1G1F1F1D2E1F1G1G1G2F1E1D1C5e3d2c1A6d3
    d2c1e1c4G1A2G1A1A1G1f1f1d2B1G1A1A1G2F1G1E1D1C8";
    REM Cielito Lindo
280 REM*************
290 M$="G3E1G3E1G1A1G1F1E1G2G1C1C1C1D1E1E1E1E1
    D1D1D1E1D3P1G3E1G3E1G1A1G1F1E1G2G1C1C1C1D1E1E1
    E1C1D2C1B1C1P1c1B1A6c1A1G6C1B1C1C1C1D1E2E1C1D2C1B1
    C1P1c1B1A6c1A1G6E1D1C1C1C1D1E2E1C1D2C1B1C4P4";
    REM SHE WORE A YELLOW RIBBON
300 REM*************
310 SCALE$="C1D1E1F1G1A1B1c4P4":SCALE
320 REM*************
330 DEF PROCdisplay
340 CLS
350 VDU 19,3,RND(3),0,0,0
360 VDU 19,0,RND(7),0,0,0
370 PRINT TAB(RND(20),RND(26))"** Music maker **"
380 ENDPROC
```

code To play the first song, enter — as a direct command — A$ = Z$ then follow this by GOTO 80, when you will be asked how fast you want *Cielito Lindo* to be played

To get song two, enter A$ = M$ followed by GOTO 80 To get your BBC Microcomputer to play scales, enter SCALE$ = A$, then GOTO 80 You can easily store tunes you've worked out in strings in this way.

00ZX81

on ZX81

The following program produces coded messages that are extraordinarily difficult to crack For example, try and crack this message

5 = AKY – 3R2Y5IKL

The only way to decode the above message is by using the following program and knowing what the keyword is Every message can be uniquely coded, and another advantage is that all spaces are coded and like letters will appear differently coded Therefore, it is no good trying to count up the most used letter or graphic symbol and calling that E or whatever, it won t work

The program works by taking your code word — say for example 'Enigma' and adding it to your message text, deducting 63 where the addition yields a final total of more than 63 The keyword is re-cycled every time For example

ENIGMA ENIGMA ENIGMA
NOW IS THE TIME

Therefore the spaces have G, E M added to them, and the two Es, for example, will appear differently One will have G added and the other I

Enter the program as listed except for line 8070 enter *all* of it in normal video Save it on tape by typing as a direct command RUN 8070 The program will then automatically run on subsequent loadings All you have to do is enter your keyword when prompted, then your message

The message can already be in code or in a decoded state just answer line 150 relevant to the state of your text By choosing a particular keyword that your fellow spy will know you can then publish your message No one will be able to decode it unless they happen to know the particular keyword

00ZX81
by Barry Cornhill

```
5 REM B.CORNHILL
10 PRINT TAB 10;"Z-CODE"
12 PRINT "FIRST ENTER YOUR KEYWORD THEN
   TEXT FOR CODING OR DE-CODING"
15 PRINT ,,"ANY KEY TO CONTINUE"
16 IF INKEY$="" THEN GOTO 16
20 CLS
25 PRINT "KEYWORD PLEASE "
30 LET Z$=""
40 INPUT K$
45 DIM A(LEN K$)
50 FOR N=1 TO LEN K$
60 LET A(N)=CODE K$(N)
70 NEXT N
80 LET T=1
90 PRINT "INPUT TEXT PLEASE"
100 INPUT M$
110 CLS
120 PRINT M$
130 GOSUB 3000
140 CLS
150 PRINT "(C)ODE OR (D)ECODE"
160 GOTO 160+(10 AND INKEY$="C")+(140 AND INKEY$="D")
165 GOTO 150
170 FOR N=1 TO LEN M$
175 IF T>LEN K$ THEN LET T=1
180 LET X=A(T)+CODE M$(N)
190 IF X>63 THEN LET X=X-63
200 LET Z$=Z$+CHR$ X
210 LET T=T+1
220 NEXT N
230 CLS
240 PRINT Z$
250 GOTO 3000
300 FOR N=1 TO LEN M$
310 IF T>LEN K$ THEN LET T=1
320 LET X=CODE M$(N)-A(T)
330 IF X<0 THEN LET X=X+63
340 LET Z$=Z$+CHR$ X
350 LET T=T+1
360 NEXT N
370 CLS
380 PRINT Z$
390 GOTO 3000
3000 POKE 16418,0
3010 PRINT AT 22,0;"(C)OPY OR (N)EXT MESSAGE ?"
3020 IF INKEY$="" THEN GOTO 3020
3030 POKE 16418,2
3040 IF INKEY$="C" THEN LPRINT Z$
3050 GOTO 20
3060 STOP
3070 SAVE "Z-CODE"
3080 GOTO 10
8000 POKE 16418,0
8010 PRINT AT 23,0;"CORRECT Y OR N ?"
8020 IF INKEY$="" THEN GOTO 8020
8025 POKE 16418,2
8030 IF INKEY$="N" THEN GOTO 90
8040 RETURN
READY.
```

Sound & vision



Making music on the Atom

The Acornsoft Atom Synthesiser is a tidy little software package for the Acorn Atom home computer Like many other Atom software packages it requires an expanded Atom — in this case 5K of program RAM and 6K of graphics RAM

My initial thought that Synthesiser was too grand a name for such a package hasn t changed, but it does use some synthesiser concepts in creating its four voices

The program comes on a standard cassette in the usual Acornsoft type packaging, which is an example to all manufacturers It takes a while to load, like all Atom cassettes, but once loaded you are assured there are no mistakes in the program because of the checksum routines that do not allow loading of 'duff' cassettes After loading there is a prompt (M/R/P/E/S/L/T ?) which serves as a kind of menu the letters referring to the available options

The first of these options is manual mode — hitting M makes the Atom keyboard into a kind of synthesiser-type keyboard, allowing you to play the keys just like a piano Of course the keyboard is not anything like as good as even the cheapest real synthesiser keyboard And it is jolly hard work hitting the right key each time

The second option is more interesting, and to my way of thinking the justification for the program The record option allows you to store a tune of up to 255 notes in the memory of the computer

When R is pressed followed by another letter to indicate which of four possible tunes you want to store the screen goes white and musical staves appear

Tunes are entered on to the empty staves via the old keyboard arrangement, rests being input via the space bar Incorrectly entered notes can be deleted using the delete key

The P or play option does just that plays one of the tunes in memory It may be obvious to you by now that the Atom Synthesiser is badly named what it should have been called is the Atom Sequencer

A sequencer is a device, often digitally or even computer based which can either memorise a tune in its own right, or memorise a tune input via a synthesiser.

E is the edit option, which is a useful function Save will store the composed masterpieces on tape, so they can be reloaded and replayed at a later date Load is the instruction to do that, and the last choice on the menu is T for Tempo

The Tempo can, and often does vary from the sedate to the frenetic

As I mentioned, I don t really think that this can be taken very seriously as a synthesiser but it is an incredibly cheap tool for anyone interested in composition

The Atom Synthesiser is available from Acornsoft, 4a Market Hill Cambridge CB2 3NJ **Sam Blythe**



How to join a magic circle

You just can t get good circles on a small computer You need a resolution of well over 500 points to get something that even begins to look like one Never mind — the world doesn t consist of circles either, except the ripples on a pond when a pocket computer is thrown in

This week we ll look at a simple 'circle' program for the BBC machine, and then try to think of one or two things to do with it For, apart from the immediate satisfaction, there is little point in just doing tricksy things on a computer for their own sake is there?

You ll need to be in mode 4 or 5, and don t forget that the BBC machine has its graphics co-ordinates expressed always between 0 and just over 1,000

```
5 INPUT CENTRE(X Y) CH CV Rem i e centre
  of circle
10 INPUT RADIUS? R
20 ANGLE=2*PI/120 Rem — 120 is number of
  sides
30 C=COS(ANGLE) S=SIN(ANGLE)
40 XA=1 YA=1
50 FOR I=1 TO 121 Rem — No of sides plus 1
60 XTEMP=XA*C—YA*S
70 YTEMP=YA*C+XA*S
80 XA=XTEMP YA=YTEMP
90 IF I>1 THEN DRAW R*XA+CH R*YA+CV
   ELSE MOVE R*XA+CH R*YA+CV
100 NEXT I
```

So far so good One more boring circle but if you have a look at the program, a number of variations might emerge You can see that the circle' is in fact made up of lots of little steps round the rim

If that value of 120 at line 20 were to be replaced by 8, and the 121 at line 5 by 9 — always one more than the number of sides — then an octagon results Similarly, regular polygons of any number of sides can be made

Since you are plotting each bit of the rim of the figure, you could use that information to do other things Try inserting at line 95 an instruction to draw a line at 45 degrees from each point, to a position say 100 screen units up and along It would be something like this

```
95 PLOT 1 100 100 Rem PLOT 1 is a relative line
```

PLOT 1, in BBC Basic means 'relative to where you are now, draw a line to a point 100 (in this case) vertically and 100 horizontally away

Now you should get a wall rising from the circle Halve one of the Rs in line 90 and you ll get an ellipse Make one of the angles slightly different at line 30 — maybe 05 different — and if you make the number at line 50 much larger you ll get a spiral Try with and without the new PLOT 1 instruction Weirdness should result!

Finally, consider using the circular information but with text How about a circle of the word 'circle'?

Brian Reffen Smith

Hand & mouth



How to root for the answer

To take up where we left off last time — how does a calculator generate square roots? The core of the algorithm may be graphically represented as shown in the diagram

The above procedure will be much swifter if we calculate a one decade at a time, and thus avoid having to find a^2 and $X-a^2$ each time a is changed For instance, once the hundreds digit is found it is squared and subtracted from X before the tens digit is calculated

Let's define a as the most significant digit of $\sqrt{X}$ previously calculated, b and j as the next digit of $\sqrt{X}$ to be found and as its exponent respectively, R_a as the current remainder $(X-a^2)$, a_j as $a+(b\times10^j)$ and R_b as the portion of R_a that would be removed by adding b to a (ie $= a_j^2-a^2$)

The idea is to approach the root from below so that a is always less than $\sqrt{X}$ and the value of b is the largest possible digit such that $R_a - R_b \geq 0$ From the above definitions we find $R_b=(a+(b\times10^j))^2-a^2$ and hence the last inequality leads to the rule that b is the largest digit such that $2ab\times10^j+(b\times10^j)^2\leq R_a$ When the digit satisfying this rule is found, it is stored, the decade counter is decremented by one and the remainder redefined for the next decade

This process may be further quickened by recognising that the remainder R_b is the most frequently calculated number in the algorithm Dust out your old maths texts discover that we may express b^2 as $\sum_{i=1}^{b}(2i-1)$ and we have a short cut method of evaluating R_b This summation leads to $R_b=\sum_{1}^{b}2a\times10^j+(2i-1)\times10^{2j}$ and if we change the above inequality to $5R_a\geq5R_b$ then the last term of the right hand side for various b values is as follows

$b = 1 \quad 10a\times10^j+05\times10^{2j}$
$b = 2 \quad 10a\times10^j+15\times10^{2j}$

Guess

a

a^2

$a=\sqrt{X}$

and in general $= (10a\times10^j)+(b-1)+5\times10^{2j}$

This makes R_b very easy to calculate, particularly if you remember that the internal microprocessor is really best at adding and register shifting

Lost in a welter of mathematics? If you are and I don't blame you try and calculate the root of 54756 given a $g = 200$, $j=1$ and hence $R_a= 14756$, noting successive values of the last term of $5R_b$ **John Gowrie**



Choosing your assembler

This is the first of a three-part series about some of the questions you should ask yourself before buying an assembler program — a piece of software that allows you to write and correct machine code programs

The first question must be 'Can you understand the instruction book with the program?' If not, are you buying the program from someone who will go on answering your questions Backup is important, for these are not games programs with a limited range of keystrokes but quite complex packages with many commands

Check that the assembler use standard mnemonics or names for the source code instructions If it doesn't then I would exclude the assembler from any short list that you make

You may also find shops offering an assembler written originally for the 8080 central processor unit (CPU), for your Z80-based computer That is almost not worth buying but there may be some extenuating circumstances

The problem is twofold First, the 8080 mnemonics are different from the Z80 equivalent Instructions Second, the Z80 has an extended instruction set and an 8080 assembler, while it will work, will deny you some of the more powerful instructions in the Z80 repertoire

After this first inspection of the program you need to look at the editor This is the section of the assembler that handles the input of new source code instructions to the program and the subsequent editing of those lines eg. LDA (CURSOR) Y which are both a line of text and an instruction for a 6502 assembler

You will spend a long time working with this part of the program and you must be happy with it

Does the screen layout suit you? Is enough information displayed about the number of lines you have typed, the amount of memory space that is free and what the program is currently doing?

Does the program reject incorrect source code statements such as LDA (RATS) X

Does the editor automatically insert line numbers when you are writing new text and when you insert new lines into an existing program?

Must you type lines of code in a fixed format? If so, does the program help you by setting the cursor at the start of the next field when you press a single key?

A free format assembler doesn't care where each part of an instruction starts provided it is terminated by the correct 'delimiter' character eg a space or comma **John Dawson**



HOBHOUSE COURT, 19 WHITCOMB STREET WC2

WHY NOT TAKE OUT A SUBSCRIPTION?

You can have *Popular Computing Weekly* sent to your home for £19.95 a year. This will cover the cost of the magazine, postage and packing

Fill in the coupon below and keep yourself up to date week by week

I would like to take out .... subscription(s) to *Popular Computing Weekly*.

Name . ...

Address ....

....

....

Tel No . ...

I enclose £ ... . for .... subscription(s)

Please make cheques payable to Sunshine Publications Ltd

CLASSIFIED ADVERTISING RATES:

Line by line: For private individuals 20p per word, minimum 10 words.
For companies, traders and all commercial bodies 40p per word, minimum also 20 words.
Semi-display: £10 per single column centimetre, minimum length 3 cm. (Please supply A/W as PMT. Or supply rough setting instructions.)
Conditions: all copy for Classified section must be pre-paid. Cheques and postal orders should arrive at least ten weeks before the publication date.
If you wish to discuss your ad, please ring 01-839 1855

Here's my classified ad.

(Please write your copy in capital letters on the lines below.)

I make this .... words, at .... per word so I owe you £ ....

Name ....

Address ....

....

Telephone ....

Please cut out and send this form to: Classified Department, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2

Learning to bat with a BBC Micro

Tim Hartnell *discusses pre-program functions and the DEFine FunctioN command*

The BBC Microcomputer's dialect of Basic, in common with other Basics, contains a number of preprogrammed functions which you can use in a program, or in the direct mode As well as the programmed functions, you can create your own with the DEF FN (DEFine FunctioN) command

Here we will look at the functions which come with the Basic as well as discussing the use of DEF FN The discussion includes a program which uses a defined function to draw a picture of a bat!

General functions:

ABS — This function ABSolute, gives the value of X ignoring the sign, so that if X was −10, ABS(X) would be 10 Similarly, if X was +10, ABS(X) is still 10.

INT — The INT function gives the whole number or INTeger part of a number, giving the largest number which is not greater than X If X was 2 42 INT(X) would be 2

RND — This is used to generate a RaNDom number If X was 20, RND(X) could be 13 7 4, 20, or any whole number between one and 20 RND(1) gives a random number between zero and one For example

```
10 REM PROGRAM ONE
20 X=RND(1)
30 PRINT X
40 GOTO 20
```

SGN — This function returns the SIGN of the variable in brackets, the SIGN of the *argument* as this variable is known If X equals 20 that is, X is a positive number SGN(X) = 1 SGN(−20) = −1 SGN(0) = 0

TAB — This is the TABulating function, which moves the PRINT position across the line the number of spaces indicated by the argument of this pre-programmed function.

Thus, PRINT TAB(7) "£ will print the £ at the seventh position across from the left hand edge, while PRINT TAB(14),"£" will print it 14 spaces across

The direction down the screen can also be specified, by adding a second argument after a comma within the brackets

Thus, PRINT TAB (4 9),"£" will print a pound sign four spaces across, and nine down

EXP — This function gives the value of e raised to the power of the argument, so PRINT EXP(5) will give 148 413159

LOG — This calculates the common logarithm of a number to base 10 so PRINT LOG(X) where X is five will yield 0 698970004, whereas LN (X) yields the natural logarithm to base e, so PRINT LN(5) gives 1 60943791

SQR — This function yields the SQuare Root of a number, so when X is five, PRINT SQR(X) gives 2 23606798

Trigonometrical functions

SIN — This gives the sine of an angle in radians SIN(5) yields −0 958924274

COS — Yields the cosine of an angle in radians PRINT COS(X) where X equals five gives 0 283662185

```
10 REM BAT
20 REM (C) HARTNELL 1982
30 MODE7
40 VDU23 8202 0 0 0
50 L=0 P=11 O=17
60 DEF FNbat(B)=SQR(L×L-B×B)
70 PRINT CHR$(12) CHR$(30)
80 PRINT TAB(O P) "O"
90 REPEAT
100 PRINT TAB (16 9) "I I"
110 L=L+1
120 FOR B=0 TO L
130 H=FNbat(B)
140 PRINT TAB (O+H P+B) "•"
150 PROCa
160 PRINT TAB (O-H P+B) "•"
170 PROCa
180 PRINT TAB (O-H P-B) "•"
190 PROCa
200 PRINT TAB (O+H, P-B) "•"
210 NEXT B
220 PROCa
230 UNTIL L=11
240 REPEAT
250 PROCa
260 UNTIL FALSE
270 DEF PROCa
280 W=TIME
290 SOUND 1 15 RND (5)+240 3
300 REPEAT
310 UNTIL TIME-W>15
320 ENDPROC
```

TAN — Produces the tangent of angle X in radians, so PRINT TAN(X) where X equals five produces 1 37340077

It is likely that you won't be used to measuring angles in radians The radian is a measure of angle chosen so that pi radians equals 180 degrees This makes things much easier in certain kinds of calculations

In all the BBC trigonometric functions the argument must be in radians to make the functions work Fortunately the BBC micro has another function called RAD, which converts from degrees into radians for you

By combining this function with one of the trig functions you can enter X in degrees and work on it directly

For example

```
10 REM PROGRAM TWO
20 INPUT X
30 PRINT SIN(RAD(X))
40 GOTO 20
```

The DEG (DEGree) functions works the other way, converting angles expressed in radians into degrees

For example

```
10 REM PROGRAM THREE
20 INPUT "ANGLE IN RADIANS" X
30 PRINT X "RADIANS IS" DEG(X) "DEGREES"
40 GOTO 20
```

Defining functions

This feature allows you to DEFine functions within a program, which you can then call whenever you need to while running the program DEF FN can save space as well as time as complex calculations can be defined with a short name, and called up at will by use of this name

There are four elements of the statement which defines the function

- The word DEF
- The name of the function which consists of the letters FN, followed by the name
- The argument of the function which follows the name in brackets
- The formula, using the argument, for working out the function

Look to the final program — Bat — in which a function is defined in line 60 The function bat(B) gets the square root of the difference between the squares of two variables and in the routine 120 to 210, uses the value H (see line 130) to determine the printing positions of the dots which will draw up the bat PROCa (a procedure), defined from line 270 is there simply to slow things down, and produce some bat-like sounds

Peek & poke

Peek your problems to our address. Ian Beardsmore will poke back an answer.

IF YOU DON'T GET ANY JOY, STICK AT IT

Phil Cooper of Lyndhurst Grove Chiselhurst writes

Q I want to write some games programs that are for two players but I would like to know if a joystick is available for the ZX81?

A A joystick is available for the ZX81, it is made by Micro Gen which also makes a controller board The firm s address is given at the end Another option is to make your own *Interface*, the magazine of the National ZX80 and ZX81 Users Club has an article in issue 2 6 (February) on how to construct your own Joystick The club can be contacted at 44-46 Earls Court Road London W8 6EJ Micro Gen is at 24 Agar Crescent, Bracknell, Berkshire

I GOT A REAL FRIDAY AFTERNOON CLUNKER

Mike St Paul of Kemplay Road Hampstead writes

Q Recently I bought a ZX81 in kit form from someone who had bought it at the end of last year as an electronic project which in the end never got started I had a lot of difficulty with it, until I realised — with the help of the instructions from another kit — that there were several major errors in the Sinclair instructions and that components were missing from the kit I have not seen much mention of this in the computer press Is it just an isolated occurrence or have a lot of kits been sent back as dud? It is just that anyone lacking sufficient knowledge to spot the mistakes might find themselves left with £50 worth of useless plastic, chips and components

A This is in fact a problem that I have met before, and for once Sinclair Research acted very quickly pulling the incorrect kits off the market The problem was in fact picked up by members of the National Users Club in December and the club reported the trouble to Sinclair It is not known how many kits were sold and no doubt some have been corrected by the people who bought them But judging by the lack of response on this issue it would seem that most of the defective kits were stopped before reaching the public

BARY, YOU CAN DRIVE MY DISC (I LOVE YOU)

Bruce Edwards of East Row Edinburgh writes

Q I would like to know if there is a disc drive available for the ZX81, and where can I get it if one is available?

A The only disc drive I know to be currently available is from Macronics, whose address is below But if you've read the first issue of *Popular Computing Weekly* you will know that one is being planned by Monolith There is also the new mini floppy, which will soon be over here from Japan This is planned to be compatible with the new ZX Spectrum, and will cost about £50 So you might find it worth waiting a little while to see if anyone brings out one of these for the ZX81 Macronics is at 26 Spiers Close Knowle Solihull, West Midlands B93 9ES

WHAY'S THE POINT OF IT ALL, YOU ASK?

Kevin Emery of Lissoms Alley Stoke-on-Trent writes

Q Can you explain something for me? I entered the following line by mistake, 10 PRINT followed by RUN I then got two lines of noughts I can understand the two lines because of the comma, but why the noughts? To make things worse I tried it on a friend s ZX80, expecting to get four lines of noughts and all I got was syntax error! Why?

A This is in fact the quickest demonstration of the integer arithmetic of the ZX80, and the decimal arithmetic of the ZX81 The most important function of the character ( ) in the ZX81 Basic is to act as a decimal point Therefore when you enter PRINT the ZX81 starts looking for a number that goes with the decimal point As there is not one it sees this as PRINT nothing point nothing, which it quite reasonably interprets as nought

Because the ZX80 does not have decimal arithmetic when the command PRINT is entered, it does not make sense The ZX80 logic is expecting a pair of inverted commas, a variable or else a calculation

IS YHIS WHERE THE INTERACTION IS?

Neville Parsons of Hillingdon Road, Harrow writes

Q Is there a way of producing a similar thing to the Sinclair's INKEY$ on an Acorn Atom? I want to write interactive programs which do not need to be stopped every time you need an input Any routine which would help would be appreciated

A If you PEEK the address £B001 (using PRINT ?£B001) you'll see its value changes depending on which key you press To build up a table which you can use, enter

```
DO P ?£B001 UO
```

and try pressing different keys to see what effect this has Then you can easily assign specific responses to changes in the value of £B001

The following program to test this is suggested in the book *Getting acquainted with your Acorn Atom*

```
10 IF?£B001=127 P "ACORN
20 IF?£B001=191 P "atom
30 P            G 10
```

This will show the effect of hitting CTRL and SHIFT

HERE'S A SUM WATT I CAN'T QUITE DO

David Bale of Westcot Road Harlow writes

Q I have been given a maths question at school about turning wattage into horse power Although I can do it on paper I would like to know if it can be done on a ZX81, and if it can how? The question is this A single bar of an electric fire has impedance, and so uses 1017 kilowatts an hour How much energy in horse power does it use in one hour and eleven minutes?

A Although you say that you can do this on paper you leave out what is probably the single most important piece of information The conversion factor of Watts to horse power This is 0 00134, and given this it is possible to put together a program Try the following

```
10 PRINT  TIME IN MINUTES
20 INPUT T
30 PRINT AT 0 17  T
40 PRINT  WATTS
50 INPUT W
60 PRINT AT 2 7  W
70 LET H=((W 60) T)*0 00134
80 PRINT  HORSE POWER IS  H
```

Line 70 is the important line First, the number of Watts per minute is obtained Then this is multiplied by the duration in minutes to give the total number of Watts used Lastly this is converted into horse power

AND YOU YHINK YOU HAVE A PROBLEM?

Simon Cray of Jasons Road, Cheam writes

Q I think I have an EDITing problem Towards the end of some programs, I just can't EDIT when I want to Although I press SHIFT and EDIT at the same time all the curser does is flash The line I want to pull out stays where it is

A The Important part of your letter is where you say the end of some programs Loss of EDIT is one of the first signs that you are running out of memory It is more likely to happen on longer lines, and is soon followed by the line you are inputting moving up the screen To be honest there is nothing much that you can do about it unless you expand your memory

Send your questions to Peek & Poke, Popular Computing Weekly *Hobhouse Court 19 Whitcomb Street, London WC2 7HF*

Competitions

Cheque book puzzle

The other morning I called at my local bank to cash a cheque. Shortly afterwards I bought a newspaper for 20p and on checking the money left in my pocket I discovered that I had twice the amount of money that I drew on the cheque.

As I had no money at all when I left home I realised that the cashier in the bank must have mistakenly transposed the amount of pounds for pence and pence for pounds when cashing my cheque.

How much was the cheque for?

Solution to Puzzle No 1

The volume of the tank for a given value of X can be found from the formula

Volume = (10 − 2 * X) * (10 − 2 * X) * X

In solving this puzzle it is logical to assume (as is indeed the case) that in progressing from the value when X is very small to the value when it is at a maximum (ie — 5 inches) the volume gradually increases to a maximum point before beginning to get smaller.

In the program below the value of X is first set at a minimum and the volume worked out. This volume is then compared with the preceding volume to see if it is either equal or less. When it is equal this marks the maximum value.

```
10 LET X = 0.0005
20 LET Y = 0
30 LET V = (10 - 2 * X) * (10 - 2 * X) * X
40 IF V < = Y THEN GOTO 100
50 LET Y = V
60 LET X = X + 0.0005
70 GOTO 30
100 PRINT X
110 STOP
```

In order to verify that there is only one maximum value the procedure can be reversed. In this case a high starting point for X is given in line 10

```
10 LET X = 5 - 0.0005
```

and this value is *decremented* in line 60

```
60 LET X = X - 0.0005
```

The results show that the answer lies between 1.667 and 1.666.

After running the program through once then the starting value of X can be re-defined as 1.666 and the steps by which it is incremented can be made smaller.

The answer is in fact 1.6 recurring.

Winner of Puzzle No 1

The winner is Paul Reynolds, Longfield, The Common, Sissinghurst, Kent, who gets £10.

Solution to Crossword No 1

Across 6 Graphic output 8 Miff 9 Inhale 11 Eskimo 12 Rave on 13 Gaucho 14 Ship 16 Speedier motor

Down 1 Traipse 2 Spifficate 3 Silicon chip 4 Motherboard 5 Gun 7 Television 10 Monitor 15 CPU

Winner of Crossword No 1

The winner is J R de Boer, Coolgardie Avenue, Chigwell, Essex, who receives £10.

Rules

Readers can enter any or all of our competitions but please use a separate envelope for each as this helps our judges.

The winner for the crossword and the winner of the puzzle will be the first name out of the hat (in each case).

Closing date for both the crossword and the puzzle is the Monday three weeks after the cover date.

Please mark your envelope 'CROSSWORD' or 'PUZZLE'.

Crossword No 5

ACROSS

1 Pat former officer (8)
6 Type of order (4)
9 Counted in days, not long to the end (6)
10 Poverty of writer turning before railway (6)
11 Arranges the smells about right (6)
13 Pet example, sing about urges (4 2)
15 Look at the sinking sun bottom (6)
16 Beast in reign of new value set (6)
18 Move slowly in Switzerland (4)
19 The best Pete return for an initial instruction (6 4)

DOWN

2 Mock up about computer products (7)
3 One in my currency (5)
4 Speechless for love of an elephant! (6)
5 Meteor rises with first half of 13A too (3)
6 Stranger ferreting about after replacing Tritium with oxygen (9)
7 Heark! The way up by chip is lonely (9)
12 Sort of loop through the window — hi Lionel (2 5)
14 Digital fasteners? (5)
15 Tracks the Basic program statements (5)
17 This kinetic sport contains itself (3)

